windream

EMC Centera Implementation

windream **GmbH, Bochum**

Copyright © 2006 - 2007

by
windream GmbH
Wasserstr. 219
44799 Bochum

Issue: 12/06 – 1.0.0.0

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form (print, photocopy, or any other form) or by any means without the explicitly written permission of **win**dream GmbH.

Important Note
All information and technical specifications in this book have been collected by the authors with great care. They cannot, however, either guarantee, take legal responsibility for or make any other warranty for the results from the use of this information.

We would also like to point out that all software and hardware logos and names are the exclusive property of the trademark, brandname or patent right holders. These are naturally protected by the appropriate laws and trade agreements.

Table of Contents

Configuration – windream **for Centera** **1**

Requirements 2
- Software - windream Server and Centera 2
- Centera application profile for windream 2
- Requirements for data replication 3
- Notes on using multicluster failover configuration 4
- End of archiving period (windream life-cycle) and archiving period on a cluster (retention period) 4

Configuration - survey 6
Configuration in detail 8
- 1. Application profile and PEA (Pool Entry Authorization) 8
- 2. Copying files to the windream directory 9
- 3. Creating entries in the Windows registry 9
- 3.1 Parameters for authorization / authentication 9
- 3.2 Parameters for the log files 12
- 3.3 Parameters for Centera settings 14
- 3.4 Parameters for backup files 16
- 4. Determining aliases (hosts) 17
- 5. Creating an ObjectStore in windream 18
- 6. Testing archive connections using WRITest.exe 23

Backup for data replication 25
- Requirements for backup 27
- Backup crawler (CNTClean.exe) for data replication 27
- Starting the backup crawler (CNTClean.exe) 28
- Parameters of CNTClean.exe 29

Log files of libmmarcCentera.dll 31
- CenteraDLL.log 31
- CenteraSession.log 32

Example of a registration file 34
Short description of the WRITest.exe program 37

Configuration – windream for Centera

Connecting the **win**dream document management system to the Content Addressed Storage (CAS) system Centera of EMC requires a program library, which is named `libmmarcCentera.dll`. This library (incl. further program and configuration files) is placed in a subfolder named `Optional` of the **win**dream program directory when **win**dream is installed. In **win**dream version 4.0, the files are directly installed in the **win**dream program directory (no subdirectory required any more).

All configurations required for using **win**dream with the Centera cluster must be made manually. For this, the files must first be copied to the program directory, and then the settings on the **win**dream server must be adjusted.

The present documentation serves as a guideline for the installation and configuration of the system. This should be managed by a system administrator, as administration rights are required and especially knowledge in the Centera and **win**dream systems.

Notes on further documentations

Detailed descriptions of the system requirements as well as of the installation and configuration of the **win**dream server can be found in the respective documentations and online help files provided by **win**dream GmbH (e.g. **win**dream server description and user manual).

Detailed information on system requirements, on the installation of the EMC Centera cluster and clients as well as on the configuration of a Centera cluster can be found in the documentation which is installed with the Centera administration tools (e.g. in "Centera Product Guide, P/N 069001221", "System Operator's Guide, P/N 069001126", Centera API Reference Guide", P/N069001185 or "Centera Programmer’s Guide", P/N 069001127").

Requirements

In order to operate **win**dream with a Centera cluster, a completely configured and ready to operate EMC Centera system (EMC Centera Cluster, client with access to the cluster and – if required – a second cluster for data replication) as well as a configured **win**dream server must be available.

Software - windream Server and Centera

- **win**dream Server version 3.6.0.13 or higher.
- **win**dream Libmmarc Centera 3.6.0.6 (Centera 3.1.477.0) patch; the patch contains the `libmmarcCentera.dll`, version 3.6.0.6, which has been created with Centera SDK 3.1.477.0. Centera V3.1 SDK is compatible to CentraStar TM version 2.0.

Note: We urgently recommend to install the patch.

The `libmmarcCentera.dll` has been tested on the following Centera cluster versions:

- libmmarcCentera.dll 3.6.0.5 - Centera Cluster 3.0.0-542-557-8615
- libmmarcCentera.dll 3.6.0.5 - Centera Cluster 3.1.0-956-654-12195
- libmmarcCentera.dll 3.6.0.6 - Centera Cluster 3.1.1-1207-703-13222

Centera application profile for windream

For **win**dream, a Centera Application Profile must exist which allows the **win**dream services and programs the following access rights (server capabilities) on the Centera cluster:

- Write
- Read
- Exist

Further information on this can be found under "1. Application profile and PEA (Pool Entry Authorization)", p. 8 and in the Centera documentation ("Product Guide, P/N 069001221").

By default, the **win**dream services use the user ID of the **win**dream mainuser to log on to the Centera cluster. **win**dream transfers the user ID (without domain designator) and his password. If this user name / password shall be used for Centera, you will be able to use it when creating the application profile on the Centera cluster.

If a PEA-file (PEA=Pool Entry Authorization) shall be used instead, the file must be made available to the application server (**win**dream

server). The **win**dream `libmmarcCentera.dll` must be configured in a way that the **win**dream services receive the logon data from this file.

Further information on this can be found under "1. Application profile and PEA (Pool Entry Authorization)", p. 8, "3.1 Parameters for authentication /authorisation", p. 9 and in the documentations of Centera ("Product Guide, P/N 069001221").

Requirements for data replication

Required is a Centera system with one cluster for archiving and a second one for data replication. Detailed information on possible topologies and configurations can be found in the documentations of Centera.

Description of the win**dream backup strategy when replicating data with Centera**

If a document is archived on the Centera cluster by the DMS-service, it is not clear yet when it will be replicated on the replication cluster. Because of this, the `libmmarcCentera.dll` can create a local backup of the document. A separately operating process (`CNTClean.exe`) will check whether the document has been replicated and whether the backup has been deleted or whether it has been moved to a "NotReplicated" directory.

The backup strategy can be configured as "Always", "Never" or "Centera depending".

Further notes on this can be found under "3.4 Parameters for the backup files", p. 16 and "Backup for the data replication", p. 25.

Note: The `CNTClean.exe` program is not part of the software package in a standard installation process of **win**dream server 3.6.0.15. However, it will be installed with patch "`libmmarcCentera_3_6_0_6_SDK_3_1_477`". Please note that the installation of this patch is urgently recommended.

Data replication directories for the backup files

The directories for the backup files named `BackupRootDir` and `NotReplicated` must be created. You create these directories on the **win**dream computer or on another drive in the file system. Both directories must be on the same level and should not be created in the root directory (see also "3.4 Parameters for the backup files", p. 16).

Sufficient storage capacity for the backup files

Sufficient hard disk capacity must be available for the copies of all **win**dream documents which shall be stored. These backup files will

only be deleted by the backup crawler `CNTClean.exe`, after data have been replicated (see also "Backup for the data replication", p. 25).

Scheduled task for the backup crawler

If data is replicated, the backup crawler must be started regularly. For this, a respective scheduled task must be configured in the Windows task manager on the **win**dream server (see also "Starting the backup crawler (CNTClean.exe)", p. 28).

Notes on using multicluster failover configuration

If a second cluster is configured for multicluster failover, the **win**dream `libmmarcCentera.dll` can be configured in a way that **win**dream will automatically establish a new connection (FPPool_Open) on the second cluster (if the first one fails) to archive data on the second cluster (see parameter `WriteRetryCount`, p. 15).

If, during reading operations, **win**dream is unable to address the Centera cluster or if objects cannot be found there, the application will try to find the involved objects on the replication cluster.

Further information on "multicluster failover" can be found in the Centera documentation (e.g. "Centera Programmer’s Guide", P/N 069001127").

End of archiving period (**win**dream **life-cycle) and archiving period on a cluster (retention period)**

In the registry on the **win**dream server, a common and identical retention period can be defined for all blobs (C-Clips) to be archived on a Centera cluster. After this period has expired, the documents can be deleted from the Centera cluster. Note that the "unlimited retention period" cannot be changed on a Compliance Edition Plus Centera Cluster. This means that the retention period will never expire and the documents cannot be deleted.

The retention period for archived documents can be set in registry entry `RetentionDays`. Default is an unlimited retention period (see p. 14). However, you can also use the standard Centera cluster settings (`RetentionDays=-2`) or a fixed retention period.

A respective deletion process will be run by the **win**dream DMS service on the respective Centera clip after the life-cycle period has expired (`FPClip_Delete`), if the document is removed from the **win**dream database. This requires the "DELETE" right in Centera.

However, if the retention period has not expired yet, this will fail.

This error will be ignored by the DMS server and the document will remain in Centera.

Note: The determination of a retention period on the Centera cluster is independent of the life-cycle administration in **win**dream. If the retention period for documents on the Centera cluster ends before the archiving period in the life-cycle has expired, it will be possible to delete documents from the cluster, or the documents will not be accessible for **win**dream any more although they (the blob references) are still available in the database.

Configuration - survey

Required adjustments of the Centera administration configuration:

1. Create the application profile for **win**dream (**win**dream mainuser) with the required access rights on the cluster (write, read and exist). If desired, create a PEA-file (Pool Entry Authorization) and make it available to the **win**dream server (see "1. Application profile and PEA (Pool Entry Authorization)", p. 8).

Required adjustments of the windream **configuration on the win**dream **server:**

2. Copy all files from the `Optional` directory to the **win**dream installation directory (e.g. to `C:\Programs\windream GmbH\windream`). The `Optional` directory is a subdirectory of the **win**dream program directory. In **win**dream 4.0 or higher, this step is not required any more.

3. Configure the local Windows registry on the **win**dream server for `libmmarcCentera.dll` (see "3. Creating entries in the Windows registry", p. 9). This requires two steps:
   - Adjust file `libmmarcCentera.reg` (`C:\Programs\windream GmbH\windream`).
   - Execute file `libmmarcCentera.reg`.

4. If required, enter the assignments (aliases) for the Centera Cluster Access Nodes to be used in file "hosts", if no local DNS is used for name resolution. You can also use the IP addresses of the access nodes for the connection of the **win**dream ObjectStore to Centera (see "4. Determining the aliases (hosts)", p. 17).

5. Create the ObjectStore in the **win**dream storage administration of the **win**dream Management Console. You must:
   - Create a new ObjectStore in the storage management of the **win**dream Management (command *New>windream Storage*), and enter the designator for the access node of the cluster as *Server* and the cluster port in field *Connection* (default `3218`). As *Archive provider*, you enter the `libmmarcCentera.dll`. See also "5. Creating an ObjectStore in **win**dream", p. 18.

6. Test the archive connection (**win**dream ObjectStore to Centera cluster) using the `WRITest.exe` program (Windream Required Interface Test). The program is installed with the software in **win**dream version 3.6.0.13 or higher (directory `Optional`).

- Start the `WRITest.exe` program and enter the `libmmarcCentera.dll`. With this program, you e.g. test the connection of **win**dream to the Centera cluster and the creation, storing and fetching of blobs. Further information on this can be found in chapter "6. Testing archive connections with WRITest.exe", p. 23.

Configuration in detail

1. Application profile and PEA (Pool Entry Authorization)

The data for logon to a Centera cluster are determined in the PEA-file for each application. The data for logon to a Centera cluster are determined in the PEA-files for the respective application. These files contain the required information for authentication and authorization (e.g. the password for the login and the access rights on the cluster).

First, an application profile is created on the Centera cluster for the respective application. Then, the entries defined in the profile are exported to a PEA-file. The file is copied to the application computer.

In order to connect to the Centera cluster, the application (**win**dream `libmmarcCentera.dll`) uses the API function named `FPPool_Open`, which is started with the parameters for the IP-address of the cluster and with the name (path entry) of the PEA-file. If this PEA-file cannot be found, the file which is determined in the system environment variable `CENTERA_PEA_LOCATION` will be used. If this file is not existing, or if it has not been set, the DLL will use the `Anonymous` Profile.

By default, the **win**dream services use the user ID of the **win**dream mainuser for logon to the Centera cluster. **win**dream transfers the user name (without domain designator) and the password. Use this ID when creating the **win**dream application profile.

The **win**dream mainuser who logs on to the Centera cluster requires the following access rights (server capabilities):

Right	**Description**	**Required for** win**dream**
Write (w)	Writes to a C-Clip	Yes
Read (r)	Reads a C-Clip	Yes
Delete (d)	Deletes a C-Clip	Yes, if the retention period is limited (see "RetentionDays", p. 14).
Exist (e)	Checks the existence of a specified C-Clip	Yes, if `CNTClean.exe` (backup crawler) is used and if it checks whether the documents archived are already existing on the replication cluster (see "Backup for data replication", p. 25).
Privileged	Deletes all copies of the	No

Right	Description	Required for windream
Delete (D)	C-Clip and can overrule retention periods.	
Profile driven Meta data (P)	Allows the storage and retrieval of additional meta data on a C-Clip's CDF on a per-profile basis.	No
Query (q)	Queries the contents of a cluster. If set to Enabled, C-Clips can be searched or in the cluster using a time-based query.	No
Litigation hold (h)	Verifies that retention hold is enabled.	No

Table 1: Centera server capabilities

Further information on this can be found under "3.1 Parameters for authentication / authorization", p. 9 and in the Centera documentation.

2. Copying files to the windream directory

This step is not required any more in **win**dream version 4.0. See also item 2. in the "Configuration - survey", p. 5.

3. Creating entries in the Windows registry

The `libmmarcCentera.dll` library is configured via entries and keys in the local Windows registry (**win**dream Server). In order to create these entries, you will first have to adjust file `libmmarcCentera.reg` according to your system environment. The file must then be executed.

An example file can be found in chapter "registry file example", p. 34.

How to adjust the Windows registry

1. Open file `libmmarcCentera.reg` in a text editor and make the desired settings.
2. Execute file `libmmarcCentera.reg`.

3.1 Parameters for authorization / authentication

The **win**dream services and applications which access Centera via `libmmarcCentera.dll` must authorize their Centera access. Additionally and depending on the requirements, rights are required

which authorize certain applications (**win**dream DMS service or `CNTClean.exe`) to execute operations on the Centera cluster (e.g. writing, reading or deleting data).

This authentication is either executed via a user ID (user name / password of the **win**dream mainuser) or via the entries of a PEA file (PEA=pool entry authorization). A respective file must be created by a Centera system administrator and must be made available on the application server (**win**dream Server). The path to the file must be determined in `libmmarcCentera.reg` or in the Windows registry on the client (see below: parameter `PeaFile`).

Parameters `AuthorizationPolicy` and `PeaFile` determine the way the authorization of the **win**dream services and the applications shall be handled.

Examples for possible configurations

1. A PEA-file for DMS service and `CNTClean`

The information of a single PEA-file available on the **win**dream server shall be used for the **win**dream DMS service and `CNTClean.exe.`

```
AuthorizationPolicy = (dez)32 (hex 0x20)
```

The path of the file must be determined in registry entry `PeaFile`, e.g. `PeaFile` = `C:\Programs\windream GmbH\windream\windream.pea`.

2. Different PEA-files for DMS service and `CNTClean`

The entries of two different PEA-files which are both available on the **win**dream server shall be used for the **win**dream DMS service and for `CNTClean.`

```
AuthorizationPolicy = (dez)20 (hex 0x14)
```

The path of the file which is used by `libmmarcCentera.dll` must be entered in registry entry `PeaFile`, e.g. `PeaFile` = `C:\Programs\windream GmbH\windream\windream.pea`.

The path of the file for `CNTClean.exe` must be entered when starting the program ("Parameters of CNTClean.exe", p. 29).

3. Always log on with user ID (name/password)

The logon of the **win**dream DMS service and of `Clean.exe` is always executed via a user ID (user name/password).

```
AuthorizationPolicy = (dez)2 (hex 0x2)
```

Note: The **win**dream DMS service always uses the ID of the **win**dream mainuser for logon.
For the backup crawler `Clean.exe`, a user ID must be entered when it is started ("Parameters of CNTClean.exe", p. 29). For this, a respective user account must have been created.

4. Logon with user ID (DMS service) and PEA-file (CNTClean.exe)

The logon to the **win**dream DMS-service is always executed via a user ID (user name/password). `CNTClean.exe` uses the entries of the PEA-file which is entered as start parameter at runtime.

```
AuthorizationPolicy = (dez)6 (hex 0x6)
```

Note: The **win**dream DMS-service always uses the ID of the **win**dream mainuser for logon.
For the backup crawler `Clean.exe`, the file for `CNTClean.exe` must be entered ("Parameters of CNTClean.exe", p. 29).

AuthorizationPolicy

The parameter determines which authentication and authorization data of **win**dream or of `libmmarcCentera.dll` (**win**dream DMS service or the `CNTClean.exe`) shall be used for access to the Centera cluster. There are different variations of authentication, and combinations of options are possible, too. The options most frequently used are `32` (`AUTH_POLICY_PEAFILE_ALWAYS`) or `2` (`AUTH_POLICY_USER`).

We recommend to use option "32".

1 - AUTH_POLICY_ANONYMOUS_ALWAYS
The logon is anonymous, user name and password are ignored. This setting is the default presetting. **Note:** Also use this setting if you want to use the PEA-file which is determined in the Windows environment variable `CENTERA_PEA_LOCATION`.

2 - AUTH_POLICY_USER
User name and password are used for logon. The **win**dream DMS service uses the ID of the **win**dream mainuser. User name and password must be transfered as parameters for `CNTClean.exe` on startup.

4 - AUTH_POLICY_PEAFILE
`LibmmarcCentera` uses the string passed as password in the API call as PEA-file, but only, if the user name is empty at the same time. Additionally, you can define via `AUTH_POLICY_PEAFILE_DEFAULT` that the PEA-file entered in "PeaFile" is used by default, if no password is forwarded. Because the DMS service always passes the mainuser / password to the DLL, this setting is only recommended if a separate PEA-file shall be used additionally. E. g. for `CNTClean.exe.` Generally, this is not the case.

8 - AUTH_POLICY_ANONYMOUS_ALLOWED
Also allows an anonymous logon (user name and password need not be entered or are ignored). E.g., this option can be set in addition to `AUTH_POLICY_PEAFILE_ALWAYS` (`32`). Recommended e.g. for tests initiated by `WRITest.exe`, which shall connect to the Centera cluster without user name / password.

16 - AUTH_POLICY_PEAFILE_DEFAULT
`LibmmarcCentera` will only use the entries of parameter `PeaFile`, if no password is passed. `CNTClean.exe` uses these entries, if no other logon data (PEA-file or user name / password) is passed as parameter on startup.

32 - AUTH_POLICY_PEAFILE_ALWAYS
The PEA-file determined in `PeaFile` will always be used. Additionally, you can also allow an anonymous logon (AUTH_POLICY_ANONYMOUS_ALLOWED (`8`)). Further combinations will not be possible then.

PeaFile
Complete path to the PEA-file with the authentication information for logon to the Centera cluster. By default, no file is determined (`PeaFile=""`). If a PEA-file shall be used, it must be available on the **win**dream server (see also "1. Application profile and PEA (Pool Entry Authorization)", p. 8).

3.2 Parameters for the log files

We will now describe the parameters for the creation of the log files. Examples of log files generated can be found under "Log files of libmmarcCentera.dll", p. 31).

Log files

Log files (parameters `DLLLogFile` and `SessionLogFile`) should be saved in the default log directory of **win**dream (`windream\reporting`).

DLLLogFile

The log file of `libmmarcCentera.dll` (`CenteraDll.log`).

SessionLogFile

The log file of the Centera session (s) - `CenteraSession.log`. The file contains detailed information on the Centera connection.

LogLevel

For testing purposes, you can increase the log level (parameter `DLLLogLevel` and `SessionLogLevel`), e.g. to `7` (`doing`). During operation, the level should be `1`.

The log level can be switched on the fly with the following action:

1. Set the entry in the registration to the desired level.
2. In the **win**dream Management Console storage administration, you create a new object type for which you determine `libmmarcCentera.dll` as provider (command *New>windream Storage*). Do not save!
3. Enter a name, select option *ObjectStore*, enter *Server* (designator for the access point of the cluster) and *Connection* (port of the cluster, default `3218`) and determine `libmmarcCentera.dll` as archive provider.
4. Refesh the "Pool list".
5. Close the dialog with *Cancel* without saving the ObjectStore settings!

DLLLogLevel

The settings of the log level and of the backup configuration are evaluated on each start of "`MMAListConfiguration (MMA_CONF_POOL_LIST)`" (refreshing the pool list).

Level
1 = Errors (default)
2 = Info (some more information)
4 = Doing (detailed information, entering and leaving functions are logged)
8 = Critical Sections (critical section or mutex (enter, wait, leave) are logged)

SessionLogLevel

The level settings are read while creating a new session. The settings of the log level and the backup configurations are evaluated on each start of "`MMAListConfiguration (MMA_CONF_POOL_LIST)`" (refreshing the pool list).

Level
1 = Errors (default)
2 = Info (some more information)
4 = Doing (detailed information, entering and leaving functions are logged)
8 = Critical Sections (critical section or mutex (enter, wait, leave) are logged)

3.3 Parameters for Centera settings

The configurations described below should be discussed with a Centera system administrator, as Centera-specific settings are determined with these parameters. The default settings can be applied as described here. After a new installation of **win**dream, the settings must be adjusted again.

Thus, we recommend to take up these settings explicitly or to save them.

RetentionDays

After having archived objects on a Centera cluster, these objects (C-Clips) can only be deleted by other applications after a defined period of retention. Parameter `RetentionDays` determines the retention period of the objects after archiving in the Centera pool. Possible entries are:

Value	**Meaning**
>0	Retention period in days.
0	No retention period; the objects can be deleted immediately after archiving.
-1	Unlimited retention period (default: `FFFFFFFF`).
-2	The default setting of the Centera cluster is used (`FFFFFFFe`).

Further information can be found in the Centera documentation (e.g. under "Retention periods and retention classes" in "Centera Programmer’s Guide", P/N 069001127").

Note: Please note that the determination of the retention period on the Centera cluster is independent of the life-cycle settings in **win**dream. If the retention period for documents on the Centera cluster ends before the archiving period in the **win**dream life-cycle administration has expired, it may be possible that documents are deleted on the cluster or that they are not accessible to **win**dream any more, although the documents (the blob references) are still available in the **win**dream database.

WriteRetryCount

If the connection to the first cluster is interrupted during archiving operations, **win**dream will again reconnect (`FPPool_Open`) to archive data on the second cluster. The number of writing attempts (repetitions) are determined via parameter `WriteRetryCount` (default is value `10` (`0xA`)). For this, a second cluster must have been configured as failover (multi-)cluster and the options for failover security (write failover) must have been set on the cluster.

Further information on this can be found in the Centera documentation (e.g. in the "Centera Programmer's Guide", P/N 069001127").

CollisionAvoidance

windream objects are archived in Centera pools as C-Clips (binary data (blobs) and associated application-specific metadata). In this context, a unique ID is always generated for each object (Centera blob), which is the Content Address (CA). All IDs of the blobs of a c-clip are saved in an XML-file. This file is also saved together with the binary and other metadata as c-clip.

In order to ensure that the content address is always unique, parameter `CollisionAvoidance` can be set. The parameter controls whether a "unique" content address is always be calculated (`1`) or not (`0`).

Value	Meaning
0	Switched off (default)
1	Switched on; a unique content address is calculated for each blob of a c-clip.

Further information can be found in the Centera documentation (e.g. "Centera API Reference Guide", P/N069001185 or "Centera Programmer's Guide", P/N 069001127).

CollisionAvoidanceThresholdKB

This parameter determines the threshold for the file size, from which a unique content address is calculated. If parameter `CollisionAvoidance` has value `1`, the settings made here will not be considered.

Value	Meaning
0	File size will not be checked.
>0	File size in kilobyte; the file size will be checked and a unique content address will be calculated during the archiving process for all files which are bigger than the size determined. Default is value 100 MByte =`102400` KByte (`00019000`).

3.4 Parameters for backup files

The following parameters determine whether backup files shall be created for blobs archived in Centera and where these backups shall be saved.

BackupRootDir

This is the directory for the backup files. In this directory, `libmmarcCentera.dll` creates subdirectories in which a copy is created in a separate directory for each blob saved in Centera. The designators of the subdirectories are created according to pattern `YYYY` from the current year (e.g. `2006`). The directories for the files are constructed as follows: `\MM\DD\HH\nnnnnnn_nnn.[n]`.

In order to generate backup files, parameter `BackupPolicy` must be set (value `1` or `2`). See also description of "BackupPolicy".

The directory (`BackupRootDir`) must be created manually (on the **win**dream server or on another accessible volume). We recommend to create the directory not as a root directly, e.g. as `C:\CenteraBackups\BackupRoot` and NOT as `C:\BackupRoot`.

Enough hard disk capacity must be available for the backup directories and, depending on the configuration, additional capacity for the objects which have not yet been replicated.

If a Centera cluster has been configured for data replication, a separate process (backup crawler `CNTClean.exe`) can check regularly whether the objects saved in Centera have already been replicated and will then delete the respective backup directories. Blobs which have not yet been replicated will be moved to directory `NotReplicated`. This directory must be on the same level as the backup root directory, e.g.: `C:\CenteraBackups\NotReplicated`.

Note: In this direcory, only those files and directories may exist which are created by `libmmarcCentera.dll` during backup processes or by `CNTClean.exe` during the deletion of replicated objects.

A detailed description of the backup process can be found under "Backup for data replication", p. 25 of this documentation.

BackupPolicy

This setting determines whether backup files are created in the backup directories during the archiving of objects or not.

The entry should contain value `2`. In this case, the backup files will be created at the exact point of time at which the current Centera cluster supports data replication.

If the value is 1, the backup files will always be created, but no information on the replication cluster will be saved in the `info.txt` at the time of archiving (see "Backup files for data replication", p. 25).

Value	Meaning
0	No backup
1	Always backup
2	Backup only if a Centera cluster has been configured for replication and if the current Centera cluster supports data replication. This is the default setting.

4. Determining aliases (hosts)

During the configuration of the **win**dream ObjectStore, the Centera access nodes via which the Centera cluster (Centera pool) is accessed must be entered as server name. In this context, Centera supports the use of names for the connections or the IP-addresses of the Centera access nodes.

If no IP-addresses shall be used, the alias names for the Centera cluster access nodes must be entered in file `hosts` on the **win**dream server. E.g.:

```
Centera1 212.3.248.35
Centera2 212.3.248.36
Centera3 212.3.248.37
```

Or, the names are resolved via a local DNS. As server name, the IP-address of the Centera access node can also be entered alternatively when creating the **win**dream ObjectStore.

The hosts configuration is not required in any case, but it serves for a better overview.

5. Creating an ObjectStore in windream

An ObjectStore is configured in the **win**dream Management Console. This ObjectStore references a Centera pool which is accessible via one or several Centera access nodes. A Centera pool consists of one or several Centera clusters. Thus, the access nodes may also belong to different clusters. However, this is not recommended by EMC[2]. Further information on this can be found in the documentations of Centera.

During the configuration of the ObjectStore, the designators for the access nodes of the cluster are entered for the *Server*, and in field *Connection* the port number of the cluster (default `3218`). As *Archive provider* you enter `libmmarcCentera.dll`.

When using several access nodes which belong to a common cluster, the complete Centera pool address (incl. the port number) must be entered as server name (field *Server* in the dialog box). See also "Possible entries for the access node of the cluster", p. 20.

The entry of the port number in the pool address is not required, if the Centera standard port (`3218`) is used.

For connections, Centera also supports the use of the IP address of the Centera access node. This way, it is also possible to enter the IP-address of the access node.

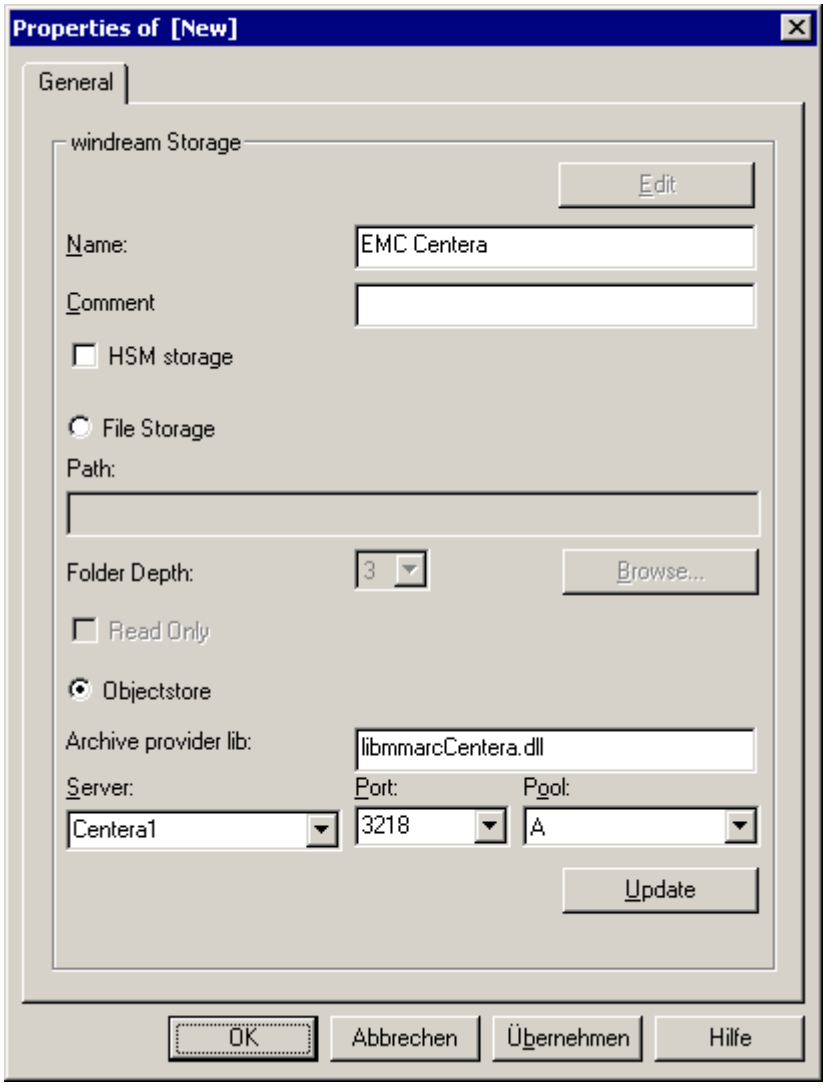


Figure 1: **win**dream ObjectStore for connecting to Centera

How to configure the windream ObjectStore

1. Start the **win**dream Management Console on the **win**dream server.
2. In the storage management on node `windream Storage>All windream Storages` you select command *New>windream Storage*.
3. Enter a *Name*, select option *ObjectStore* and enter in field *Server* the designator(s) of the access node(s) of the cluster. In field *Connection* you enter the port of the cluster (default `3218`). If the port is part of the access node designator, you enter in field *Port* value `0` (see also "Possible entries for the access node of the cluster", p. 20).
4. As *Archive provider*, enter `libmmarcCentera.dll`.
5. Refresh the pool list (button *Refresh*).
6. Select entry 'A' from the pool list.

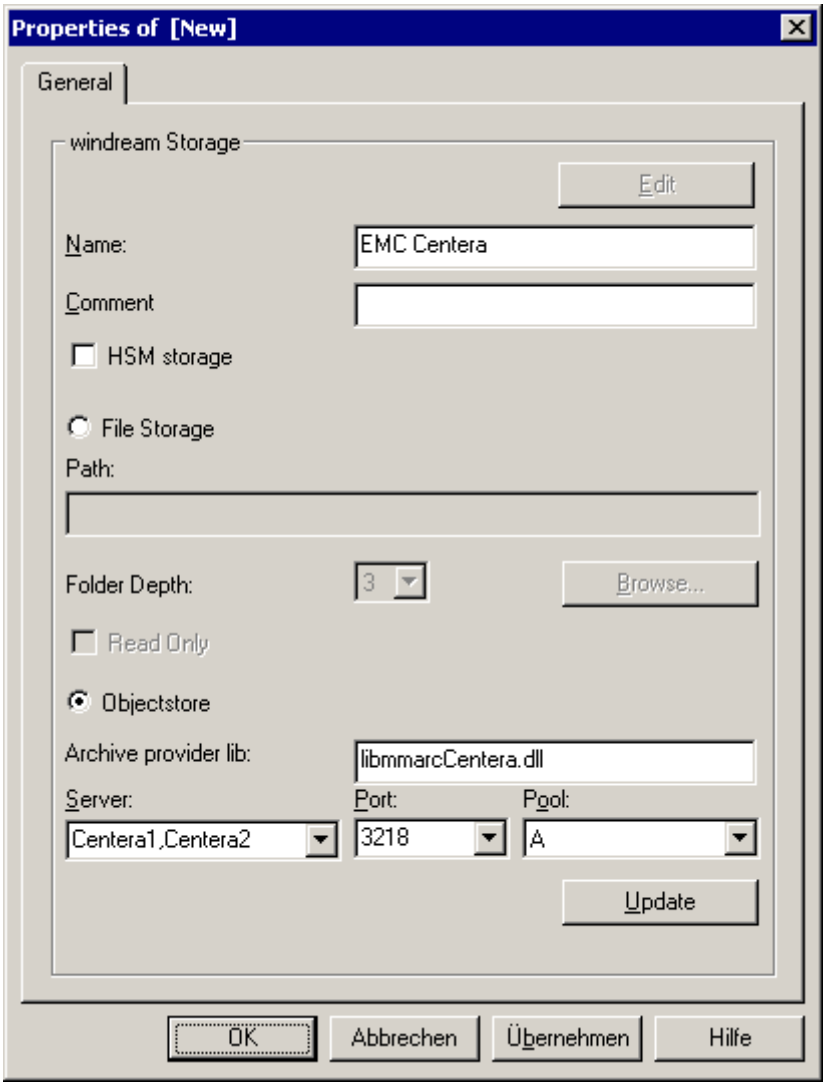


Figure 2: **win**dream ObjectStore – several access nodes with standard port

7. Close the dialog box by clicking *OK* or *Apply* to save the settings.

Note: A **win**dream pool is not required for archiving on a Centera cluster. However, on reasons of compatibility, a pool designator is required. For this, the system generates and uses pool designator "A".

Possible entries for the access node of the cluster

As server name, you enter the Centera pool address. This address may consist of a designator or of a designators list of the access nodes of the Centera cluster pool. Instead of the alias name, you can also determine the IP addresses of the access nodes.

Pool address syntax of the Centera cluster

Excerpt from Centera documentation "Centera API Reference Guide, P/N 069001185"

```
...

The format is:

<pooladdress> is a <hintlist>
<hintlist> ::= <hint> ("," <hint>)*
<hint> ::= <ipreference> [":" <port>]
<port> ::= tcpport number (defaults to 3218)
<ipreference> ::= <dnsname> | <ip-address>
<dnsname> ::= DNS name is a DNS maintained name that resolves
to one or more IP addresses (using round-robin)
<ip-address> ::= 4-tuple address format

A hint is a single pool address and a hintlist contains one
or more hints.

...
```

Further information can be found in the Centera documentation.

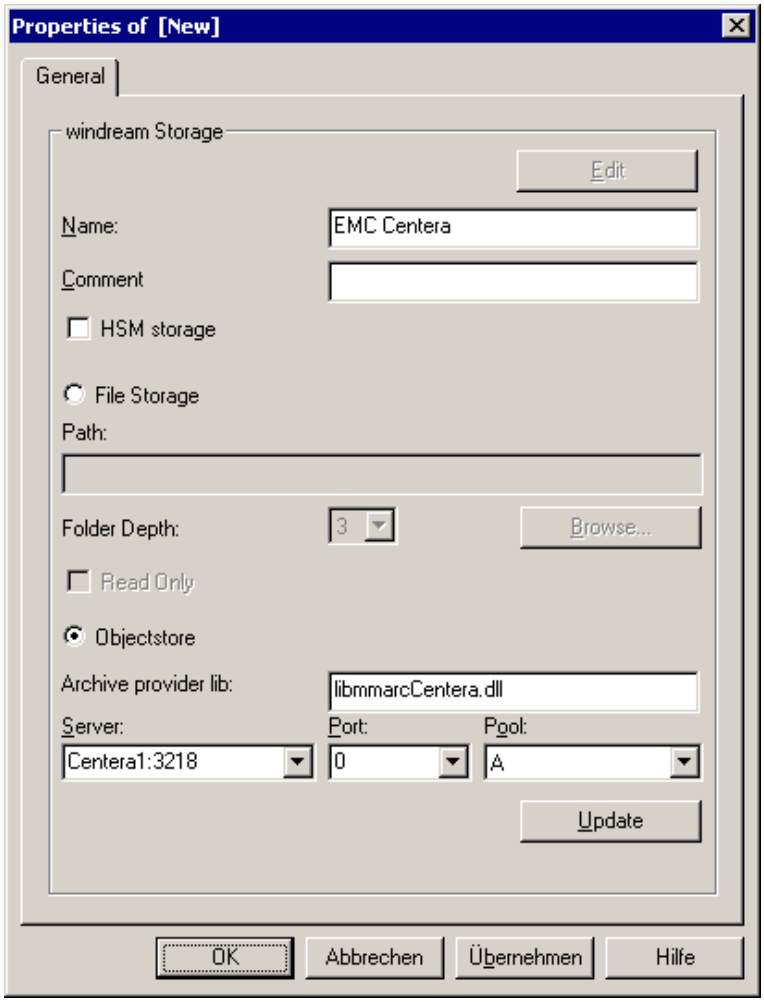


Figure 3: **win**dream ObjectStore – access node with port number in the server name

Examples for entering the pool address (server name)

When configuring the ObjectStore, you enter the pool address in field *Server*. The entry in field *Port* depends on the port number to be used and on the syntax of the pool address.

Server name	Port	Meaning
Centera1	0 or 3218	An access node and standard Centera port `3218`
Centera1:3218	0	An access node and port in the name (e.g. standard Centera port `3218`)
Centera1,Centera2	0 or 3218	Two access nodes and standard Centera port `3218`
Centera1:3218,Centera1:3219,Centera1:3220	0	Three access nodes with different port numbers.

Table 2: Pool address for connection to Centera cluster

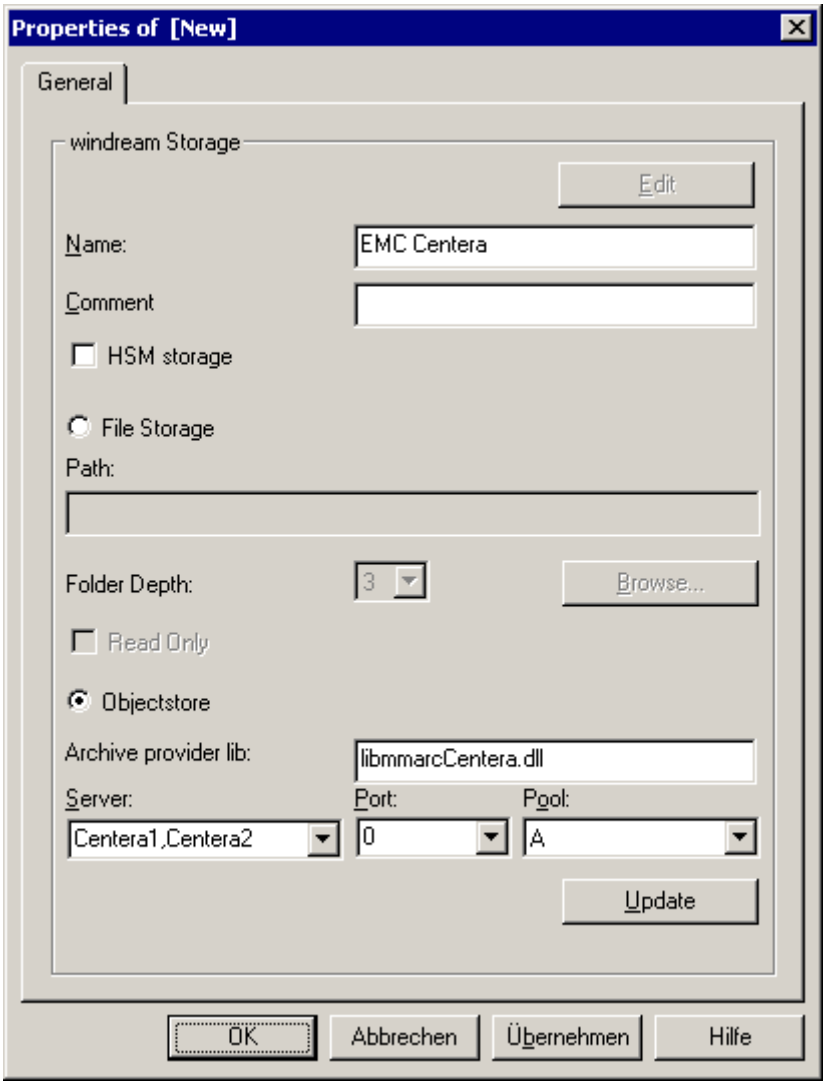


Figure 4: **win**dream ObjectStore – two access nodes with standard port

6. Testing archive connections using WRITest.exe

With program `WRITest.exe` (WRI=windream required interface for archive connections) of **win**dream GmbH you can test the currently configured connection parameters (between **win**dream and Centera cluster).

Further information on the `WRITest.exe` program can be found under "Short description of WRITest.exe", p. 37.

Start the program from the **win**dream program directory. Load `libmmarcCentera.dll` as archive provider (in sector *Library Select*) and enter the desired connection data (*User/Password*, *Server/Port* and *Pool*) in sectors *Session Logon* and *Session Access*. With button *Startup New*, the connection will be established and a session will be opened. The *Session ID* and all other information on actions / tests are edited in the status bar. If errors occur while trying to establish the connection, the program will edit respctive messages. In this case, please check the parameters and test them once again.

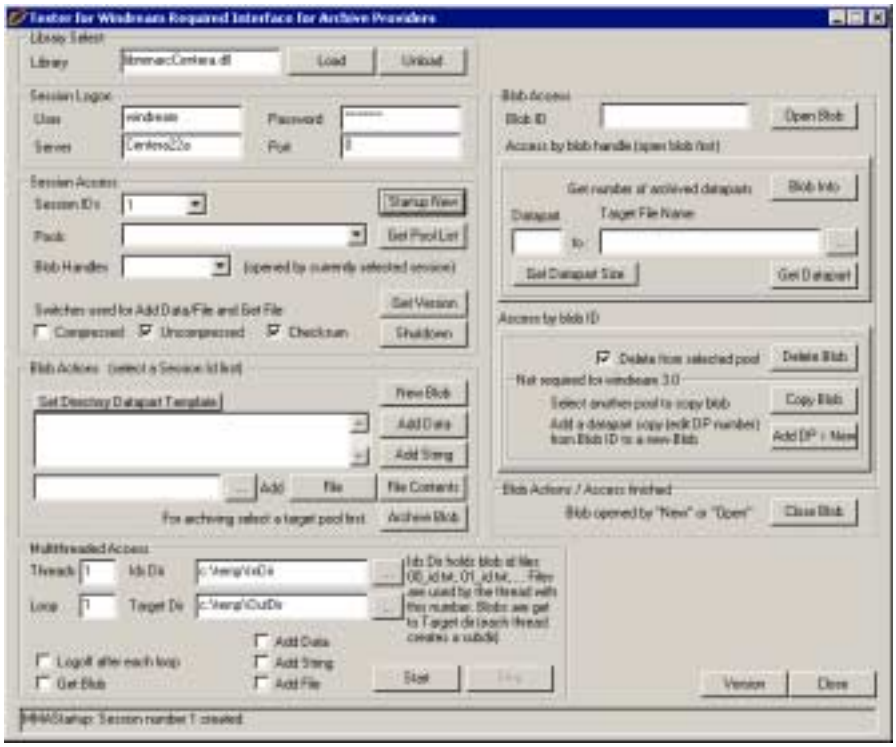


Figure 5: WRITest.exe – testing archive connections

You can also create, store, open and delete blobs on the Centera cluster (pool) with the program for test purposes. A new blob is e.g. created via button *New blob*. Via commands *Add data*, *Add string*, *File* or *File contents* you can add single data parts with respective contents to a blob. An archived blob is opened via *Open blob*. The blob-ID of the blob is displayed in the status bar. From there, it is to be copied to field *Blob ID* to open the blob. You can view information (e.g. the number of data parts) of a blob (*Blob info*) or delete a blob from the Centera pool

(*Delete blob*). Note that the deletion of a blob is only possible if the retention period set (parameter `RetentionDays`) allows this.

Further information on the `WRITest.exe` program can be found under "Short description of WRITest.exe", p. 37.

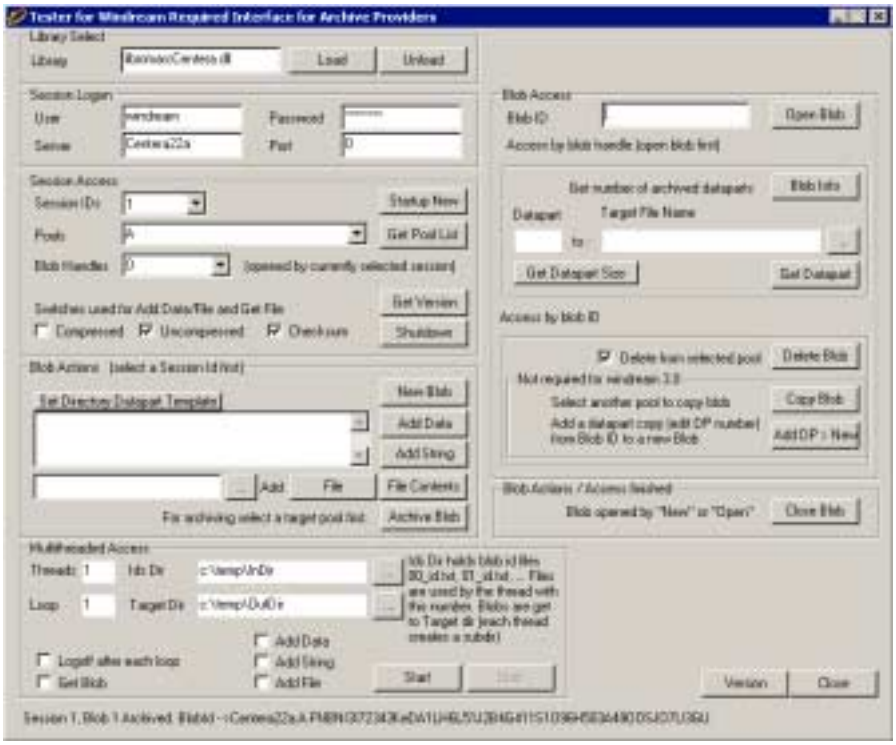


Figure 6: WRITest.exe – archiving a blob in the Centera pool

Backup for data replication

On reasons of failover security, a second Centera cluster can be configured and used for data replication.

Excerpt from the Centera documentation "Replication Guide, P/N069001168":

```
...
To support disaster recovery, you can configure a cluster to
transparently replicate all stored objects asynchronously to
a remote cluster. This complements mirroring by putting
copies of data in geographically separated sites. If a
problem renders an entire cluster inoperable, the replica
cluster can support your application server until the problem
is fixed. The speed of the replication process depends on the
bandwidth available between the source cluster and the
replica.
...
```

The backup files

In order to ensure failover security of the data replication, the **win**dream `libmmarcCentera.dll` library can save data copies of the blobs locally on the **win**dream server in a determined backup directory while archiving data. Data will only be deleted, if the corresponding Centera files (C-Clip) has been replicated on a second cluster.

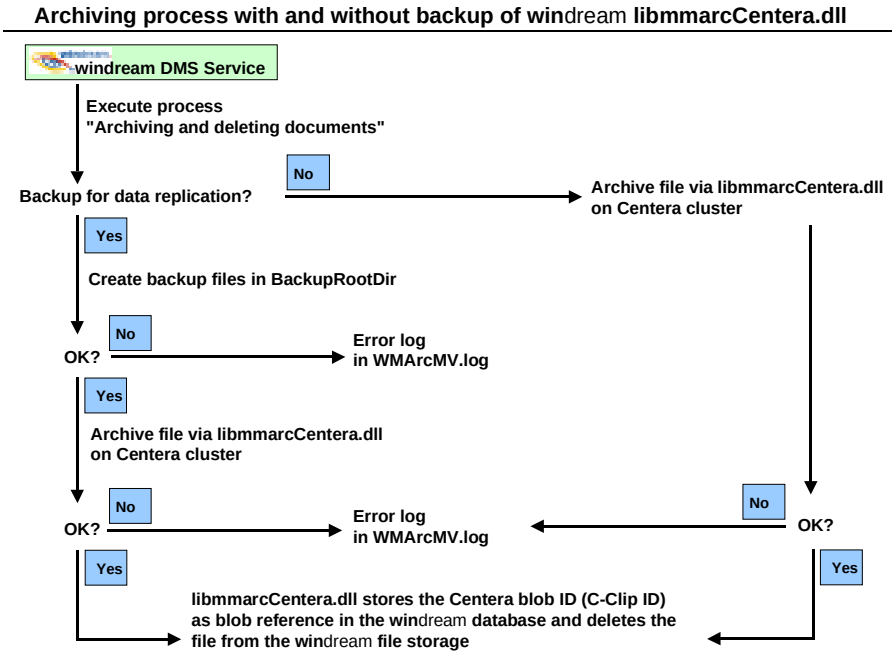


Figure 7: Archiving process with and without backup of libmmarcCentera.dll

A copy is generated for each object archived, as soon as the archiving process is initiated by the **win**dream DMS service via the **win**dream Agent (task "Archiving and deleting objects").

First, `libmmarcCentera.dll` creates subdirectories in the backup directory, in which a separate directory is created for each blob.The designators of the subdirectories are created according to pattern `YYYY` from the current year numbers (e.g. `2006`). The directories for the files are constrcuted as follows: `\MM\DD\HH\nnnnnnnn_nnn[.n]`(M=Month, D=Day, H=Hour, nnnnnnnn_nnn=Random number).

The format of the last directory is undetermined. Extension `[.n]` determines how often this documents has already been checked by the backup crawler on status "replicated" (see p. 27). Example:

C:\CenteraBackups\BackupRoot\2006\01\15\23\12345678_123.[2]

In the object directories, copies of the original file (0) and – if available – additional data parts are stored (under a consecutive number). E.g.

0 – copy of the original file

1 – a data part

2 – a data part.

Note: In the backup directory named `BackupRootDir` and in all subdirectories, only the directories created by `libmmarcCentera.dll` may exist. No other files or directories may be created here.

Info.txt files

Additionally, info files (`info.txt`) are generated in the object directories, which contain the following information on each blob in different sectors:

[Backup]	
Orig=	The designator of the Centera cluster with which the connection is established (see "5. Creating an ObjectStore in **win**dream", p. 18).
Repl=	The designator of the Centera replication with which the connection is established. Can also be empty, if e.g. no cluster has been configured for replication.
Id=	Centera C-Clip ID.

[Windream]	
Ref=	The object reference in the **win**dream ObjectStore.
DPCount=	number of data parts (e.g. 3).
DPFile=	The numbers of the data part which contains the original file, beginning with "0" (e.g. "1" or "0,1") In **win**dream, this is generally one data part only.

Example of an info.txt file

```
[Backup]
Orig=centera22a
Repl=128.221.200.63:3218,128.221.200.61:3218,128.221.200.62:3
218,128.221.200.60:3218,128.221.200.63:3218,128.221.200.61:32
18,128.221.200.62:3218,128.221.200.60:3218
Id=64FDM6P9LBJUAe3E3GV8MTQ4KOSG4115MMLR9L012R68HNGOBDVIA
[Windream]
Ref=centera22a.A.64FDM6P9LBJUAe3E3GV8MTQ4KOSG4115MMLR9L012R68
HNGOBDVIA
DPCnt=3
DPFile=2
```

Centera blob-ID as blob reference in windream

After having created the backup files, the objects will be archived. The `libmmarcCentera.dll` stores the Centera blob IDs (Content Address) as reference in the **win**dream database. Then, the files will be deleted from the **win**dream file storage sector ("Filestore").

Requirements for backup

In order to create the backup files:

- directories `BackupRootDir` and `NotReplicated` must exist (see p. 16),
- sufficient hard disk capacity must be available for the backup directories and – depending on the configuration – additionally for the objects which have not been replicated yet and
- parameters `BackupRootDir` and `BackupPolicy` must have been configured in the Windows registry (see p. 16).

Backup crawler (CNTClean.exe) for data replication

The backup crawler (`CNTClean.exe`) checks whether the objects saved in Centera have already been replicated and will then delete the respective backup directories. Blobs which have not been replicated so far will be moved to the `NotReplicated` directory.

This directory must be created on the same volume level as the backup root directory, e.g. `C:\CenteraBackups\NotReplicated` (see also parameter "BackupRootDir", p. 16).

Via parameter `-retries` (see below) you can determine how often an object shall be checked again. If it is not replicated after the last repetition, it will be moved to the `NotReplicated` directory.

Via parameter -age [DDHH] you determine the objects which shall be checked. All objects which were archived before the period determined (DDHH – days and hours) will be checked (see"Parameters of CNTClean.exe", p. 29).

If non-replicated objects of the `NotReplicated` directory shall be checked again, you will be able to move the respective subdirectories from there to the backup directory `BackupRootDir` again.

Backup crawler CNTClean.exe - data replication with local backup files

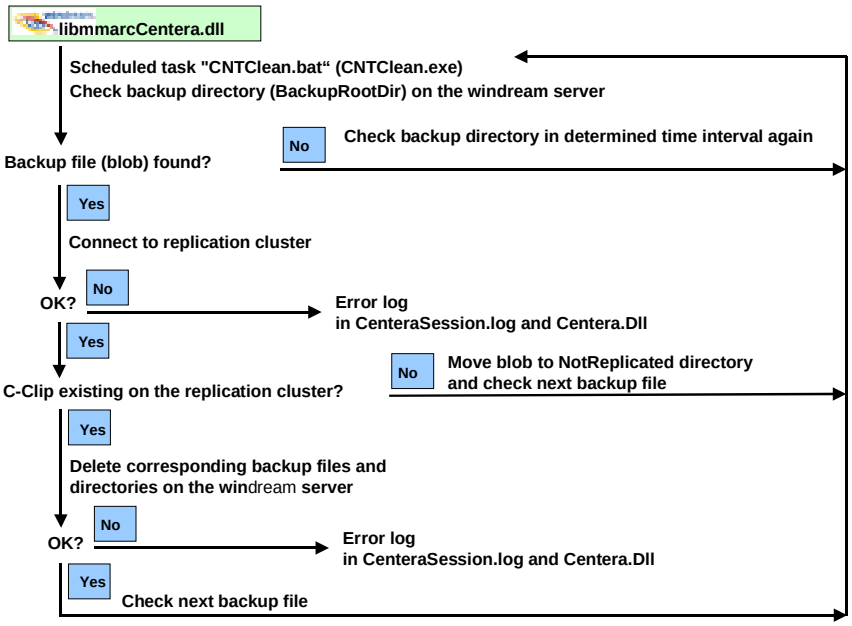


Figure 8: Backup crawler CNTClean.exe

Starting the backup crawler (CNTClean.exe)

The backup crawler has been designed as a separate program that should be started regularly via a scheduled task from the Windows Task Manager.

Thus, you create e.g.a batch file in which you determine the program start with the desired parameters. Then, you configure a scheduled task in the Windows Task Scheduler which will execute the batch file at the determined point of time.

An example of how to start the backup crawler can be found in file `CNTCleanSample.bat` and in the `CNTClean.txt` file of the **win**dream installation folder. File `CNTClean.txt` also contains a listing of the start parameters.

Example of starting CNTClean.exe

```
CNTClean.exe -action 1 -log "C:\Programs\windream
GmbH\windream\Reporting\CntCleanup.log" -srv Centera_B
-age 0200 -root "C:\CenteraBackups\BackupRoot" -retries
4 -notrepl "C:\CenteraBackups\NotReplicated" -peafile
"C:\Programs\windream GmbH\windream\windream.pea"
```

Parameters of CNTClean.exe

Parameter	Meaning
Version 3.6.0.2 or higher of Centera backup crawler	
-action <1,2>	Determines whether the backup crawler deletes the backup folders after having checked the replicated data or not (`1` = check only, `2` = delete files and folders).
-age [<DDHH>]	Determines the objects to be checked. All objects which were archived before the determined time period (DDHH – days and hours) will be checked. Default is one day: `0100`. Alternatively, the objects to be checked can also be determined via parameter `-date` (see below).
-date [<YYYYMMDDHHMM>]	An alternative of parameter `-age` for selecting the objects to be checked. All directories which are older than the determined date will be checked (YYYYMMDDHHMM – year, month, day, hour, minute).
-log <logfile>	The log file of the backup crawler
-notrepl <NotReplicated root directory>	The directory for non-replicated objects (see also parameter "BackupRootDir", p.16). **Note:** The directory must be on the same volume as the backup directory (parameter `-root`). In this directory, only files and folders may exist which are created by `libmmarcCentera.dll` for backup processes or by `CNTClean.exe` when deleting replicated objects.

Parameter	Meaning
-port [<replication cluster port>]	The port of the replication cluster; if no port has been determined, Centera standard port `3218` will be used.
-retries [#]	Maximum number of checks (default: `3`).
-root <backup root directory>	The basic directory for the backup files (see also parameter "BackupRootDir", p.16). **Note:** The directory must be on the same volume as the directory of the non-replicated data (parameter `-notrepl`). In this directory, only files and directories may exist which are created by `libmmarcCentera.dll` during backup processes or by `CNTClean.exe` when deleting replicated objects.
-srv <replication cluster>	Name of the Centera replication cluster.
Version 3.6.0.1 or higher, Centera backup crawler	
-user <user>	User name (e.g. the **win**dream mainuser); the corresponding user account must be configured and available; may not be determined, if parameter `-peafile` has been set.
-passwd <password>	Password; the corresponding user account must be configured and available; may not be entered, if parameter `-peafile` has been set.
-peafile <path to authorization pea-file>	The PEA-file from which the authorisation information is read for `CNTClean.exe`. The file must have been created and configured in advance. The entry will only be possible, if no user-ID (user/password) is determined. See also "3.1 Parameters for Authentication", p. 9.

Table 3: Parameters of the Clean.exe backup crawler

Note: Please note that the parameters in the Windows registry contain the correct entries (see also "3. creating entries in the Windows registry", p. 9 and "3.1 Parameters for authentication", p. 9.

Log files of libmmarcCentera.dll

The description of the log file settings and levels can be found under "3.2 Parameters for log files", p. 12.

CenteraDLL.log

The file is created at the position which is determined in the registry entry (default: `DLLLogFile=C:\Programs\windream GmbH\windream\Reporting\CenteraDll.log`).

Log outputs referring to the `LibmmarcCentera.dll` API are saved according to the currently set log level in the log files (parameter `DLLLogLevel`). Incoming API calls are distributed to separate sessions. The sessions use the Centera API (example see `CenteraDLL.log`).

The first value behind the timestamp determines the ThreadID which has initiated the call, e.g. `[00002216]`. By default, the registry values read are edited while loading.

Excerpt from log file CenteraDll.log

```
...
11/02/06-14:54:23 [00000968] -----------------------
11/02/06-14:54:23 [00000968] [DLL] Current module version  : '3.6.0.6'
11/02/06-14:54:23 [00000968] [DLL] Log file : C:\Programs\windream
GmbH\windream\Reporting\CenteraDll.log
(HKEY_LOCAL_MACHINE\Software\windream GmbH\Centera\DLLLogFile)
11/02/06-14:54:23 [00000968] [DLL] Log level : 7 (1=error, 2=info, 4=doing,
8=criticalSections)
(HKEY_LOCAL_MACHINE\Software\windream GmbH\Centera\DLLLogFile)
11/02/06-14:54:23 [00000968] [DLL] Session Log file : C:\Programs\windream
GmbH\windream\Reporting\CenteraSession.log
(HKEY_LOCAL_MACHINE\Software\windream GmbH\Centera\SessionLogFile)
11/02/06-14:54:23 [00000968] [DLL] Session Log level : 7 (1=error, 2=info, 4=doing,
8=criticalSections)
(HKEY_LOCAL_MACHINE\Software\windream GmbH\Centera\SessionLogLevel)
11/02/06-14:54:23 [00000968] [DLL] WriteRetryCount : 10 (HKEY_LOCAL_MACHINE\Software\windream
GmbH\Centera\WriteRetryCount)
11/02/06-14:54:23 [00000968] [DLL] Retention : -1 (days) --> '-1' (calculated seconds) (
>0=days 0=NONE, -1=INFINITE(default),
-2=CLUSTER-DEFAULT) (HKEY_LOCAL_MACHINE\Software\windream GmbH\Centera\RetentionDays)
11/02/06-14:54:23 [00000968] [DLL] CollisionAvoidance : 0 (0=OFF,1=ON) Default:0
(HKEY_LOCAL_MACHINE\Software\windream
GmbH\Centera\CollisionAvoidance)
11/02/06-14:54:23 [00000968] [DLL] CollisionAvoidance Threshold (kBytes) : 102400
(Default:102400) For files greater this size collision
avoidance will be activated. Obsolete when CollisionAvoidance = 1. Set to 0 if no size check
necessary.
(HKEY_LOCAL_MACHINE\Software\windream GmbH\Centera\CollisionAvoidanceThresholdKB)
11/02/06-14:54:23 [00000968] [DLL] AuthorizationPolicy: 1 (Default:1). 1=always anonymous
(disables all other policies), 2=by
user+password, 4=by peafile in password parameter, 8=allow anonymous, 16=default peafile,
32=always use peafile.
(HKEY_LOCAL_MACHINE\Software\windream GmbH\Centera\AuthorizationPolicy)
11/02/06-14:54:23 [00000968] [DLL] AuthorizationPolicy NOTE: If you want to use the centera
API system environment variable 'CENTERA_PEA_LOCATION', you must configure 1 (always
anonymous) here.
...
```

CenteraSession.log

The file is stored at the position which has been determined in the registry entry (default: `SessionLogFile"="C:\\Programs\\windream GmbH\\windream\\Reporting\\CenteraSession.log`).

Log outputs which refer to the Centera API (example see `CenteraSession.log`) are saved in the file according to the log level set (parameter `SessionLogLevel`).

The first value behind the timestamp determines the ThreadID, which has initiated the call, e.g. `[00001896]`.

Before the session, some Centera information is edited by default, which is traced by the respectively addressed Centera cluster while establishing a connection.

Parameter	**Meaning**
PoolRef	Internal pool reference ID
clusterID	Cluster ID of the pool (e.g.: `74cf2446-1dd2-11b2-bee7-9e82e41faa5c`)
clusterName	Cluster name of the pool
replicaAddress	Pool address (IP-address) of the cluster on which the C-Clips are replicated (e.g. `199.245.235.96`).
version	Version of the Centera pool server software (e.g. `2.2.2.338-3352`)
Capacity	Usable total pool capacity of all access nodes in bytes (e.g. `5813539799040`)
Space	Free pool storage capacity in bytes (e.g. `5772303738880`)
purge allowed	Pool capability PURGE ['true','false']
delete allowed	Pool capability DELETE ['true','false']
delete allowed	Pool capability EXIST ['true','false']

Table 4: Information on the configuration of the Centera cluster in the log file

Excerpt from log file CenteraSession.log

```
...
02/15/06-15:58:55 [00002736]  [SES] Data  ********** Opening pool: 'centera22a(anonymous)'
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo: PoolRef=317561303081000
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo: clusterID=a76ef122-1dd1-11b2-853f-
9709b17fc8bd
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo: clusterName=not configured
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo:
replicaAddress=128.221.200.63:3218,128.221.200.61:3218,128.221.200.62:3218,128.221.200.60:321
8,128.221.200.63:3218,128.221.200.61:3218,128.221.200.62:3218,128.221.200.60:3218
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo: version=3.1.0-956-654-12195
```

```
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo: Capacity=12550009651200
Space=6635654979584
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo: purge allowed : 'false'
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo: delete allowed : 'true'
02/15/06-15:58:56 [00002736]  [SES] Data  PoolInfo: exist allowed : 'true'
...
```

Example of a registration file

```
REGEDIT4

[HKEY_LOCAL_MACHINE\SOFTWARE\windream GmbH\Centera]
"DLLLogFile"="C:\\Programs\\windream GmbH\\windream\\Reporting\\CenteraDll.log"
"DLLLogLevel"=dword:00000001
"SessionLogFile"="C:\\Programs\\windream GmbH\\windream\\Reporting\\CenteraSession.log"
"SessionLogLevel"=dword:00000001
"RetentionDays"=dword:FFFFFFFF
"WriteRetryCount"=dword:0000000a
"CollisionAvoidance"=dword:00000000
"CollisionAvoidanceThresholdKB"=dword:00019000
"AuthorizationPolicy"=dword:00000001
"PeaFile"=""

; ----------------------------------------
; DLLLogFile:
;
;     the library log file.
;     contains detailed information of this library API access
; ----------------------------------------
; DLLLogLevel:
;
;   - level and backup policy settings are immediately evaluated on each
;     "MMAListConfiguration(MMA_CONF_POOL_LIST)" call
;
;     1  = errors (default)
;     2  = info (some more information)
;     4  = doing (detailed information, entering and leaving functions is logged)
;     8  = critical sections (critical section or mutex (enter,wait,leave) is logged)
; ----------------------------------------
; SessionLogFile:
;
;     the session(s) log file.
;     contains detailed information of centera access
; ----------------------------------------
; SessionLogLevel:
;
;   - level settings are read on each creation of a new session.
;   - level and backup policy settings are immediately evaluated on each
;     "MMAListConfiguration(MMA_CONF_POOL_LIST)" call
;
;     1  = errors (default)
;     2  = info (some more information)
;     4  = doing (detailed information, entering and leaving functions is logged)
;     8  = critical sections (critical section or mutex (enter,wait,leave) is logged)
; ----------------------------------------
; RetentionDays:
;
;   > 0  : retention period (in days)
;     0  : No retention (blob is deletable immediately after archiving)
;    -1  : (FFFFFFFF) infinite (default)
;    -2  : (FFFFFFFe) use centera cluster default setting;
; ----------------------------------------
; WriteRetryCount:
;
;   > 0  : (default: 0xA) number of retries for write operations when WRITE-failover
;          is possible.
; ----------------------------------------
; CollisionAvoidance:
;
;     1 : enabled (for each blob a uniqe content address is calculated.
;     0 : OFF (default)
; ----------------------------------------
; CollisionAvoidanceThresholdKB:
;
;     0 : file size is not checked
;   > 0 : for files greater than this size (KBytes) collision avoidance is enabled
;         during archiving. (default: 100 MByte = 102400 kByte (hex:00019000))
;         this setting has no effect if CollisionAvoidance: is set to 1 (ON)
; ----------------------------------------
; AuthorizationPolicy:
;
```

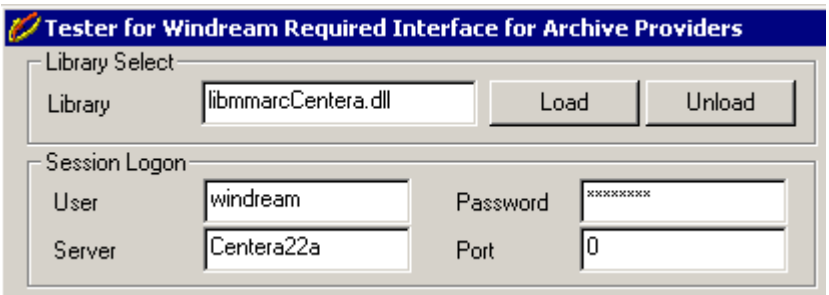
```
;       the policy libmmarcCentera.dll uses for centera pool entry authorization.
;
;  details:
;
;       1       AUTH_POLICY_ANONYMOUS_ALWAYS (default)
;               // always login anonymously (ignore user/password).
;               // this disables all other policies
;               // NOTE:
;               // use this setting if you want the environment variable
;               // CENTERA_PEA_LOCATION evaluated.
;               // you can NOT disable the evaluation of environment variable
;               // CENTERA_PEA_LOCATION.
;               // you must set it to empty instead.
;
;       2       AUTH_POLICY_USER
;               // use user/password (if both provided. This is done e.g.
;               // by windream DMS-Service)
;
;       4       AUTH_POLICY_PEAFILE
;               // Use password parameter as pea-file (ONLY if User parameter is empty).
;               // Else a default PEA-File can be used if AUTH_POLICY_PEAFILE_DEFAULT (16)
;               // is set.
;               // this Setting us just useful if a different PEA-File should be used for
;               // DMS-Service and e.g. CNTClean.exe.
;
;       8       AUTH_POLICY_ANONYMOUS_ALLOWED
;               // allow empty user/password (anonymous) e.g. for Testing connection with
;               // WRITest.exe.
;
;       16      AUTH_POLICY_PEAFILE_DEFAULT (only valid in combination with
;               // AUTH_POLICY_PEAFILE)
;               // use pea file from "PeaFile" registry value if no pea-file given in
;               // password parameter
;
;       32      AUTH_POLICY_PEAFILE_ALWAYS
;               // always use pea file from registry.
;               // this disables all other policies but can be combined with
;               // AUTH_POLICY_ANONYMOUS_ALLOWED
;
; examples:
;
;       decimal 32 (hex 0x20):
;       use "PeaFile" same for both DMS-service AND CNTClean.
;
;       decimal 20 (hex 0x14):
;       use "PeaFile" just for usage by windream DMS-service.
;       and another dynamic pea-file for CNTClean.exe (to set in CNTClean.exe parameters)
;
;       decimal / hex: 1
;       if always the system environment variable 'CENTERA_PEA_LOCATION' has to be
;       evaluated (for DMS-service AND CNTClean) then set policy to 1
;
;       decimal 12 (hex 0xC):
;       if the system environment variable 'CENTERA_PEA_LOCATION' has to be
;       evaluated for DMS-service, but CNTClean has to use a discrete
;       pea file (to set in CNTClean parameters) then set policy to
;
;       decimal / hex 2:
;       always user/password login.
;       NOTE: DMS-service will always use mainuser without domain prefix and his passwort
;             for login
;       NOTE: in this case you have to provide user/password in CNTClean parameters
;
;       decimal / hex 6:
;       use user/password login by DMS-service but a discrete pea-file by CNTClean
;       NOTE: DMS-service will always use mainuser without domain prefix and his passwort
;             for login
;       NOTE: in this case you have to provide discrete pea-file in CNTClean parameters
;
; ----------------------------------------
; PeaFile:
;
;   the pool entry authorization file that has to be used.
;   read more info in authorizationPolicy comment above.
;
```

```
;   default: "" (empty string)
; ---------------------------------------


[HKEY_LOCAL_MACHINE\SOFTWARE\windream GmbH\Centera\Backup]
"BackupRootDir"="C:\\CenteraBackups\\BackupRoot"
"BackupPolicy"=dword:00000002

; ---------------------------------------
; BackupRootDir:
;
;   Here a Path <yyyy>\<mm>\<dd>\<hh> will be created.
;
;   for each object that is backup'ed before archiving into centera
;   a subdirectory <nnnnnnnn> is created
;
;   in this subdirectory following files are stored:
;
;               - files "0", "1", "2", ... containig the archived dataparts
;
;               - "info.txt" is written with following contents
;
;       [Backup]
;       Orig=       the original centera cluster name (connection string)
;       Repl=       the replica cluster name (connection string) (may be empty)
;       Id=         the centera Clip-ID
;
;       [Windream]
;       Ref=        the returned windream objectstore reference
;       DPCount=    the datapart count (e.g. 3)
;       DPFile=     the datapart number(s) starting at 0 of those dataparts containing a
;                   file (e.g. "1" or "0,1")
; ---------------------------------------
; BackupPolicy:
;
;   0 : no backup
;   1 : always backup
;   2 : (default) only backup if centera cluster provides a replica cluster name
; ---------------------------------------
```

Short description of the WRITest.exe program

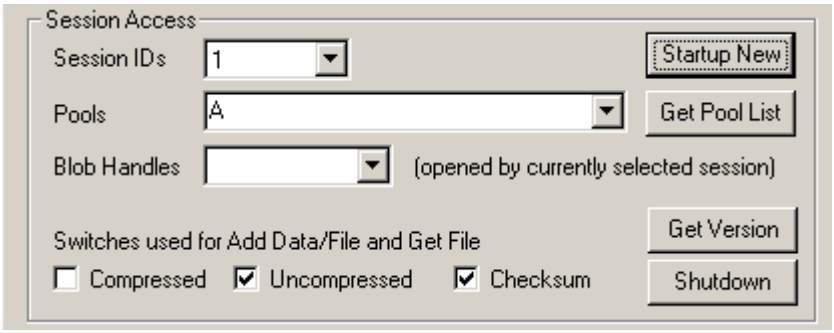
```
----------------------------------------
WRITest description.
WRI = Windream required Interface for archive connections.
----------------------------------------


Common Information:
    All output is shown in the status line at bottom of the dialog.
```

```
----------------------------------------
1. Library Select
----------------------------------------


Library: the archive library
    libmmarc.dll = windream objectstore
    libmmarcCentera.dll = centera

"Load":
    Load the library (only once at beginning of test)

"Unload":
    Unload the library (only once at end of test)


----------------------------------------
2. Session Logon (Logon Data)
----------------------------------------

User/Password:
    the user/password to connect to archive
    Specials:
        LibmmarcCentera probably uses a pea file as configured
        in centera registry. in this case user/password are ignored

Server/Port:
    the archive server/port
    Specials:
        LibmmarcCentera interprets port "0" as Centera Default port.
```



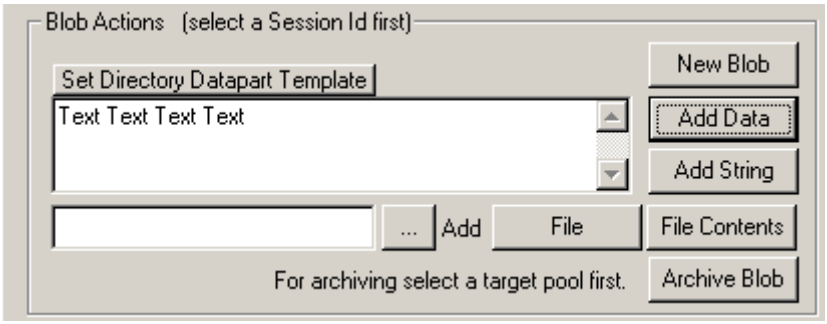

```
----------------------------------------
3. Session Access (Login a session)
----------------------------------------

"Startup New":
```

```
    open a new session (id shows up in "Session IDs")

"GetPoolList":
    shows the windream pool list
    Specials:
        LibmmarcCentera always shows a Default Pool "A"

"GetVersion":
    shows the Archive Version

"Shutdown"
    shutdown the Session (selected in "Session IDs" list) (only once at end of
test)

Options:
    Compressed/Uncompressed/Checksum:
        DO NOT CHANGE SETTINGS. For future use.
```

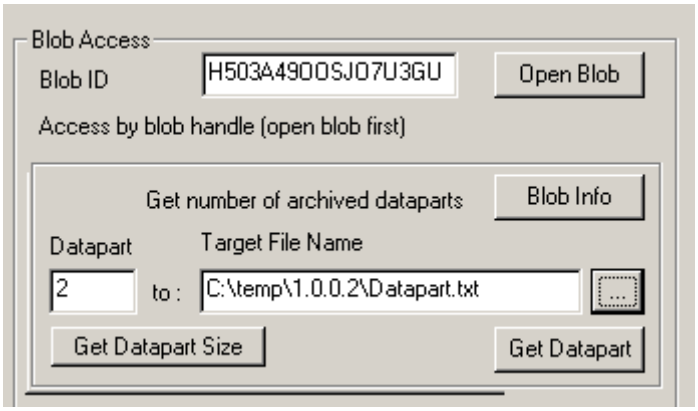



```
------------------------------------------------------------
4. Blob Actions (Archiving)
-----------------------------------------
"New Blob"
    A new Blob (Binary Large Object) is created and opened. (NOT YET ARCHIVED!)
    New Handle for this blob shows up in "Blob Handles" drop down list.

"Add Data" / "Add String"
    Add the value you entered in the edit box to the blob.
    INTERNAL:
    A sample text can be preset e.g. with button "Set Directory Datapart Template"

    NOTE: With each click on either Add Data/Add String/Add File/Add File Contents
    a new datapart is added to the new created blob.
    You my get info about number of these dataparts with
    button "Blob Info" later when executing a retrieval.


"Add File" / "File contents"
    adds a file or file contents to the blob

    NOTE: With each click on either Add Data/Add String/Add File/Add File Contents
    a new datapart is added to the new created blob.
    You my get info about number of these dataparts with
    button "Blob Info" later when executing a retrieval.


"Archive Blob"
    archives the blob
    The BlobId is shown in the status line.
    For a retrieval you must select the blob-Id with copy/paste.

    NOTE:
    A Blob that was archived by "Archive Blob" is closed automatically
```

```
    and must be reopened first for retrieval.

    Specials:
        The blob id is the last value shown in the status line.
        E.g.
            "MyServer.A.203480923840928340924"
        Make sure that you copy just this part of the blob id.
        The Blob id has always the format "SERVER.POOL.ID"
```




```
-----------------------------------------
5. MultiThreaded Access
-----------------------------------------
    Archiving/Retrieval Actions can here be tested multithreaded.


-----------------------------------------
6. Blob Access (Retrieval)
-----------------------------------------

Blob Id:
    The windream blob id of an archived object.
    E.g. "MyServer.A.120348302384092302384O"

"Open Blob"
    finds blob and opens it in the connected archive

"Blob Info"
    gets info about number of dataparts added to this blob


Datapart - Target File Name:
    "Get datapart size"
            shows the size of the requested datapart
    "Get datapart"
            gets the requested datapart (beginning with 1) to a local file
```



```
----------------------------------------
7. Access by Blob ID (Deletion/Copy)
----------------------------------------

"Delete Blob"
    tries to delete the blob identified by "Blob-ID" from the requested Pool

"Copy Blob" / "Add DP -> New"
    INTERNAL


----------------------------------------
8. Blob Actions / Access finished
----------------------------------------
"Close Blob"
    closes the blob you opened by "New Blob" or "Open Blob"
    NOTE:
    A Blob that was archived by "Archive Blob" is closed automatically
    and must be reopened first for retrieval.
```